Deep-CNN based Semantic Segmentation of Aortic Dissection Images

Sara N. Amini
Stanford University
amini00@stanford.edu

David J. Bell
Stanford University
djb218@stanford.edu

Indrasen Bhattacharya
Stanford University
indrasen@stanford.edu

Abstract

A key step in the management of type-B aortic dissection (TBAD) involves deciding when to surgically intervene on the aorta. This is based on the interpretation of follow up CT scans. In this work, we attempt to work towards developing an image-based risk prediction model. The first step in this process relies on the automated classification of the aorta into three regions: {true lumen, false lumen, background}, using deep convolutional networks for the semantic segmentation task. We evaluate and compare the performance of U-net and LinkNet, two fully convolutional architectures for semantic segmentation. We find validation set Dice coefficient exceeding 0.85 on the test and validation set, using a simple slice-based segmentation technique.

1 Introduction

Type B aortic dissection (TBAD) is a vascular emergency caused by a micro tear in the inner lining of the aorta. This tear permits passage of blood into the aortic wall, which under high pressure splits the wall into true and false lumina [1]. In some cases TBAD can progress to fatal aortic rupture, but definitive treatment (with either extensive open or endovascular surgery) also carries a high risk of mortality and morbidity [2]. Improved risk stratification is needed to distinguish between individuals with high risk TBAD who would benefit from preventive intervention from those who would fare better with conservative management. One of the major challenges for developing an image-based risk prediction model is segmentation. Manual extraction of morphologic features from computed tomography (CT) scans is extremely time consuming to perform on large data sets. The goal of this project is to develop a model to automatically segment aortic dissection CT images into true and false lumina. This is the first crucial step in the development of a risk prediction model to help guide management of TBAD.

2 Related work

Aortic segmentation is a challenging problem. Not only is TBAD a relatively rare diagnosis, which limits access to large datasets, but the CT appearance of true and false lumina can often resemble one another. Furthermore, there is significant heterogeneity in the appearance of lumina across scans. Li et al. are the only other group to attempt a fully automatic approach to segmentation of the dissected aorta. They trained their algorithm on 16 CT scans with a U-net based architecture, achieving a DICE score of 0.82-0.86 [3]. 3D u-net architectures have gained recent popularity for use in medical imaging segmentation. De Fauw et al. report the use of such architecture to segment retinal pathology and risk stratify patients based on their results. They were able to utilize a sparse annotation procedure which would not be generalizable to TBAD given that the behavior of aortic pathology is less predictable than that of retinal pathology [4]. Nikolov et al. report similar success with 3D u-net architecture segmenting organs at risk of damage during radiotherapy treatment. However, segmenting aortic pathology is more challenging, as whole organs are quite morphologically distinct from neighboring tissue (in contrast to the true and false lumina) and are also relatively uniform in appearance between individuals [5].

3 Dataset

Our dataset consists of 24, anonymized CT aortogram studies from the Department of Cardiovascular Imaging at the Stanford Medical Center. Each scan is made up of 800 256x256, grayscale, axial images. Corresponding ground truth images, also

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

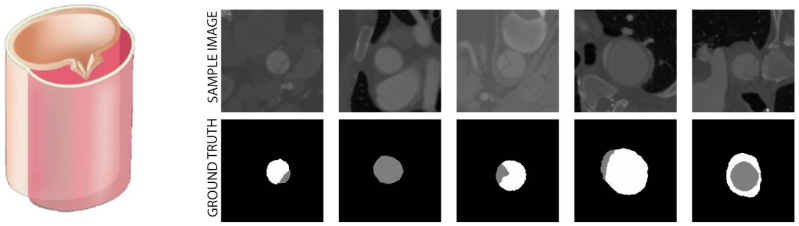

Figure 1: Left: An intimal tear leading to the formation of two aortic lumina. Right: The segmentation of true and false lumina is made difficult by significant heterogeneity in appearance across differentscans. Top row: CT images obtained by random sampling across studies and Z location, bottom row: corresponding ground truth segmentation (white: false lumen, grey: true lumen, black: background). Neither a center line nor a color difference is a sure way to discriminate lumina. Human error is low due to availability of finely sampled Z data.

256x256 and in grayscale format, had already been manually labeled, with white indicating true Lumen, gray indicating false Lumen and black indicating background.

3.1 Data Preprocessing

1400 randomly selected images from 24 different studies were chosen for training, validation and test purposes. Train, validation and test splits were 72,14,14 respectively. All input images were normalized prior to use and reformatted to 256x256x1. Ground truth images were encoded as one-hot vector which resulted in images with a 256x256x3 format, each channel representing one of three classes true lumen, false lumen, background.

In order to directly interface with the Keras Linknet encoder blocks, the grayscale images were repeated and concatenated along the channel dimension (3 channels). The images were also upsampled using nearest-neighbor interpolation to the same size as the CamVid dataset, for easier interfacing. The training/validation/test dataset split was 1000/200/200 images. The images were chosen randomly by generating a random permutation of the list of image names, and by selecting the train/val/test images serially. iPython notebooks with these utility scripts have been uploaded.

4 Architecture and Implementation

U-net is small and easily trainable. LinkNet is medium sized, partially transfer-learned (encoder weights are pretrained), and uses residual in each of the encoder and decoder blocks. This makes it fast, but also easy to train. Both these architectures are fully convolutional and learn a pixel to pixel mapping for semantic segmentation.

4.1 Metric

We chose mean dice coefficient as the metric to measure the performance of the model by comparing the model's output to the ground truth:

$$DC = \frac{2|y \cap \hat{y}|}{|y| + |\hat{y}|}, \langle DC \rangle = \frac{1}{N_c} \Sigma_c DC_c \tag{1}$$

4.2 Loss Formulation

A combination of categorical cross entropy loss and dice Loss was used for the loss function and defined as follows:

$$CCE = -\sum_{i=1}^{3} \hat{y}_i log(y_i) \tag{2}$$

$$L = w_0 CCE + w_1(1 - DC) \tag{3}$$

4.3 Network Architecture

U-net

We explored using an adaptation of U-net neural network architecture [7] (Fig. 2). U-net is an end-to-end deep neural network originally designed to perform binary segmentation of biomedical images into foreground and background. Most operations in

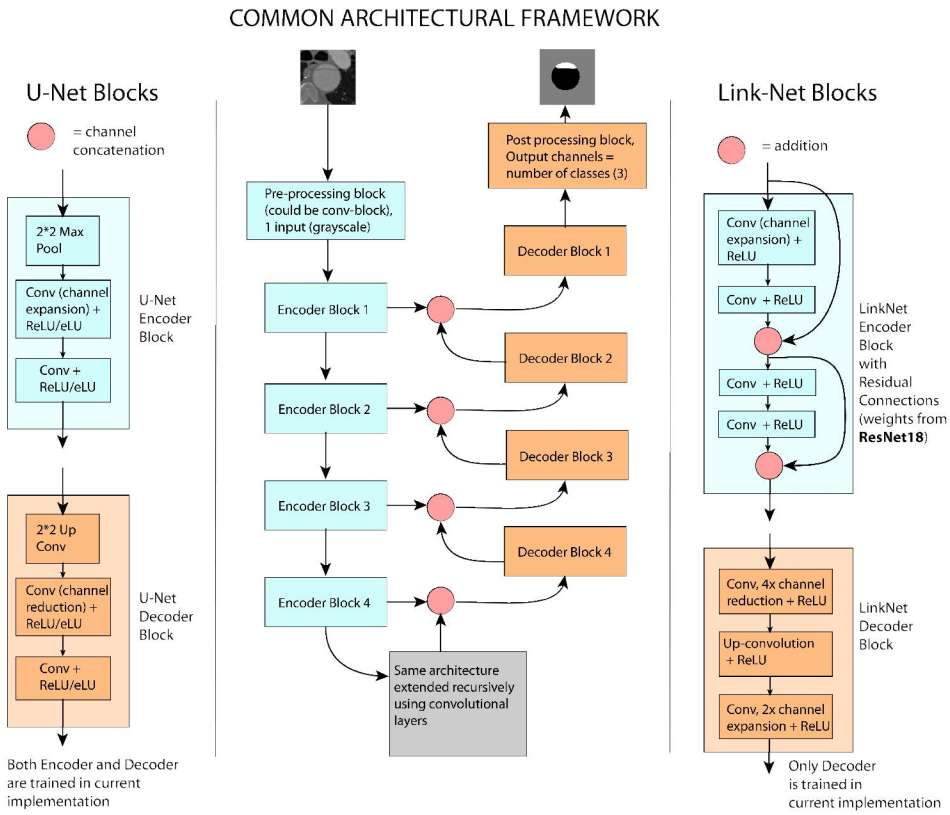

Figure 2: Comparison of U-net and LinkNet architectures for semantic segmentation. Both are fully convolutional.

the first half of the model are convolutions followed by a non linear activation function, ReLU. Max pooling is used to reduce the feature map in this part of the network. After each max pooling operation we increase the number of feature channels by two. Next the U-net uses an expansion path to create a segmentation map. This path consists of sequences of up convolutions followed by concatenations with features mapped from the contraction path. At the final step of the network, a convolution layer followed by a Softmax function is applied to map each pixel in the input image to one of the three classes. Weights were initialized using He normalization [16]. Batch size of 16 was used due to memory limitation. Batch normalization was used, along with Adam optimization for training. Combined loss function was used with w0 and w1 equal to 0.5 for training (eq. 3).

LinkNet

The LinkNet architecture is based on the Encoder-Decoder structure inherent to the fully convolutional approach (Fig. 2). The encoder block consists of four convolutional layers, with residual connections as shown. In the current implementation, pre-trained encoder weights from ResNet-18 [11] were used, with training limited to the decoder parameters. The ability to use pre-trained weights allows for greater flexibility and access to a richer set of image features for modeling the non-linear segmentation function. A Keras-based implementation [12] was cloned and used to build the model. Decoder weights trained on CamVid and Cityscape datasets are publicly available, however the model needed to be trained for the aortic dissection task. The three output channels were obtained by a simple polynomial fit of the Z dependence of each channel: $I(x, y, z) = I^0(x, y) + I^{(1)}(x, y)z + I^{(2)}(x, y)z^2$. The three polynomial coefficients were used in the three channels. This approach does help eliminate some slice to slice variation that does not have a significant effect on the segmentation result. Further, this approach was compared with just replicating the image into the three RGB channels and using that to train. A batch size of 20 was used, and some tuning of the Learning rate, L2 regularization coefficient and drop out probability was performed.

3D U-net

We also explored a 3D U-net implementation. It is worth noting that the result for this model is preliminary. To prepare data for this model we first sampled every 5th image of each study dataset. Due to similarity between sequential images this method could help to capture the changes along z-axis with only 4 images instead of 16. We then used 4 sequential images across the z-axis resulting in a 4x256x256x1 as an input to the model. The remainder of the architecture is similar to the previously

Number of Features	Learning Rate	Train	Dev	Test
3.1 e7	1 e-4	0.92	0.89	0.89
7.8 e6	5 e-4	0.90	0.88	0.88
1.9 e6	5 e-4	0.91	0.89	0.89
1.1 e7	5e-4	0.96	0.92	0.93

Table 1: Mean Dice Score For Different Implementations. The top 3 rows are UNet (25 epochs) and the bottom row is for LinkNet (early stopping, DropOut p = 0.05, L2 = 1E-5).

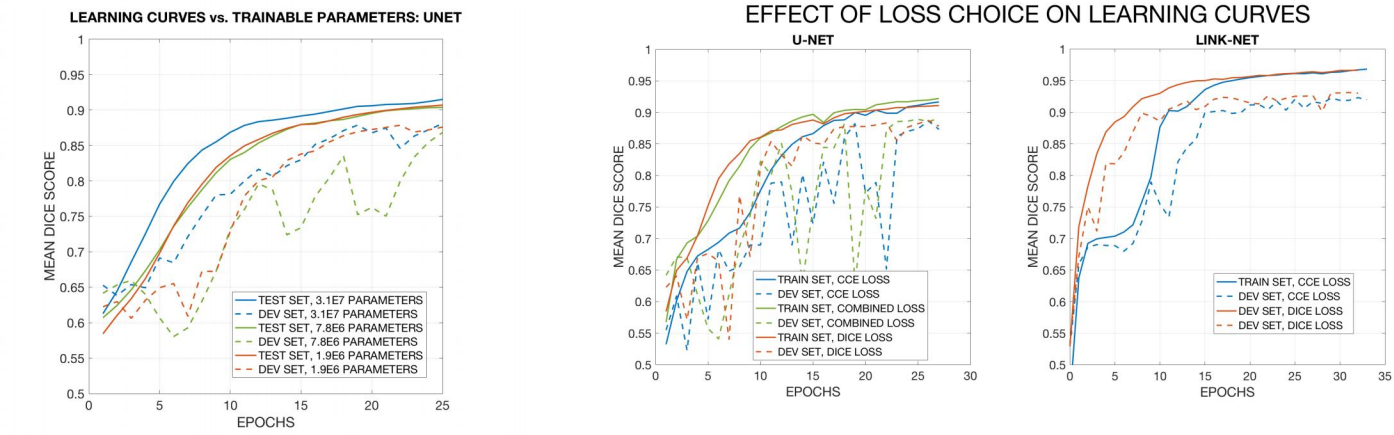

Figure 3: Left: Effect of shrinking filter channels on U-net performance. The filter channels were reduced uniformly in order to reduce the number of trainable parameters. The network could be effectively trained even with around 2E-6 parameters. Right: Choice of loss was found to be important for early convergence in both LinkNet ad UNet. However, the CCE and Dice score loss based Dice accuracy metric eventually converge to similar values. A combined loss was also found to be effective for UNet.

described U-net, however the 2D convolutions were replaced with 3D convolutions to accommodate the new axis. 3D Max pooling was not applied to the first axis and was only used to reduce feature map and second and third axis. The ground truth was generated from one-hot encoded correspondents of the 4 selected images, resulting in 4x256x256x3 masks. Batch size of 4 was used due to memory limitation. Train, validation and test split of 80,10,10 was used on 912 available sample data. Dropout was applied to help with reducing the overfitting to train data.

Training and Hyperparameter Search

In order to find the optimal number of parameters needed to train the best performing U-net model, we examined 3 different versions. The results for these models are displayed in Table 1. The models were generated by halving the number of filter channels in the convolution layers. It was noted that the model with the fewest features performed nearly as well as the model with the most. The model with the greatest number of features needed a smaller learning rate to achieve similar results to the model with the fewest parameters (Fig. 3).

A minibatch size of 10-20 images was found to be appropriate, with larger minibatch sizes being limited by memory availability. Batch normalization was used, along with Adam optimization. The ReLU activation function was used for all layers. Soft-max was used over three classes to make the prediction. Categorical cross-entropy loss (eqn. 2) was used to train the LinkNet network. The Dice coefficient was used to test performance. The Dice coefficient could also be used as a loss function. However, as Fig. 3 shows, the results eventually converge to the same accuracy. We also explored Learning Rate tuning using both Caviar and Panda techniques, and found that a Learning Rate around 5e-4 was suitable for both U-Net and LinkNet. The result is summarized in Fig. 5.

5 Results and Discussion

LinkNet and UNet based on the 2D slices produced similar performance (Validation and Test mean DICE score of 0.85 or higher on the original study). The new study had unforeseen geometries and this led to a decrease in performance (see Table 2). Polynomial fitting in Z for LinkNet did not produce significant benefits since the fit was only considering a local Z-window.

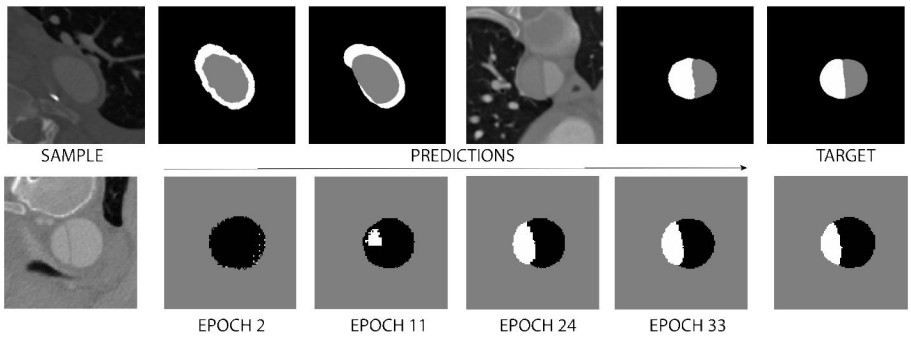

Figure 4: Top: U-net Examples, (L) Novel dataset, (R) Original Dataset; Bottom: LinkNet training sequence example showing improvement in segmentation predictions.

Model	Train	Dev	Test	Novel
U-net	0.93	0.89	0.89	0.72
Link-net	0.96	0.91	0.91	0.73
3D U-net	0.92	0.87	0.88	0.80

Table 2: Mean DICE Score

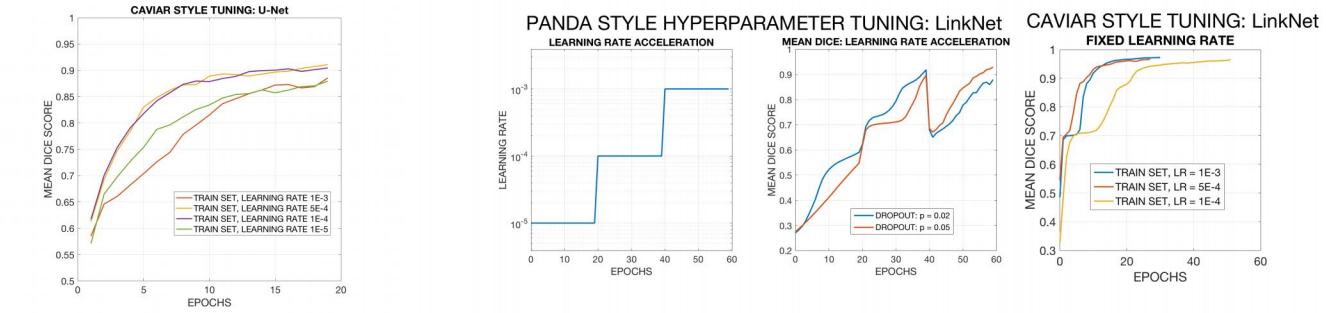

Figure 5: Left: Caviar style tuning of learning rate on UNet, best values for learning rate were around 1-5E-4. Right: Panda and Caviar style hyperparameter tuning for LinkNet. DropOut was kept fairly low at p=0.05 or less, and Learning rate was varied between 1E-4 and 1E-3. The network could be effectively trained even with 1E-3 learning rate. 1E-4 was found to be too slow. For the Caviar style Learning rate search, the drop out parameter was fixed at p=0.05.

While the 3D U-net architecture already produces a higher mean Dice score than LinkNet and U-net, it could also benefit from further hyperparameter tuning. We also concluded that Dice Loss helps speed up the training compare to CCE.

6 Conclusion and Future Work

We explored three different architectures to perform semantic segmentation of the true and false lumina and report promising results. While this is a challenging medical imaging segmentation problem, our results may be improved even further with data augmentation to decrease over-fitting. We note that high human accuracy (mean intersection over union of 0.95) is possible when the radiologist has access to contiguous Z slices that show the variation of aortic cross section. Therefore, we expect that learning a sequence model for images (such as a convolutional LSTM) might be beneficial. Another approach to incorporate 3D information may be through a 3D CNN, an extension of the 3-Dimensional approach attempted here. In addition to this, model performance might also be improved by training with the 3D sagittal and coronal views of the CT scan as well as the axial view.

7 Contributions

Sara Nasiri Amini - U-net and 3D U-net research and implementation.
David Bell- aortic dissection background, introduction, literature review and related work, R-CNN research (not implemented)
Indrasen Bhattacharya - LinkNet research and implementation
Code for both models available on GitHub at https://github.com/sara-nasiriamini/cnn-based-semantic-segmentation-of-aorta

Acknowledgements

We would like to thank Aarti Bagul for her feedback and guidance. We would also like to thank Drs Lewis D Hahn and Dominik Fleischmann from the department of Cardiovascular Imaging at Stanford Medical Center for providing access to the data and project feedback.

References

[1] DeBakey ME, Henly WS, Cooley DA et. al. Surgical Management of Dissection Aneurysms of the Aorta. J Thorac Cardiovasc Surg 1965; 49:130.

[2] Tsai TT, Nienaber CA, Eagle KA. Acute aortic syndromes. Circulation 2005; 112:3802.

[3] Multi-Task Deep Convolutional Neural Network for The Segmentation of Type B Aortic Dissection Jianning Li, Long Cao, Yangyang Ge, Cheng W, Bowen M, Wei G https://arxiv.org/abs/1806.09860v4

[4] De Fauw et. al. Clinically applicable deep learning for diagnosis and referral in retinal disease Nature Medicine volume 24, pages1342–1350 (2018)

[5] Nikolov et. al. Deep learning to achieve clinically applicable segmentation of head and neck anatomy for radiotherapy https://arxiv.org/abs/1809.04430

[6] Garcia-Garcia, A. & Orts-Escolano, S. & Oprea, S. O. & Villena-Martinez, V. & Garcia-Rodriguez, J. (2017) *A Review on Deep Learning Techniques Applied to Semantic Segmentation*, https://arxiv.org/pdf/1704.06857.pdf

[7] Ronneberger, Olaf & Fischer, Philipp & Brox, Thomas. (2015) *U-Net: Convolutional Networks for Biomedical Image Segmentation.* Springer, Cham, 2015.

[8] Chaurasia, Abhishek & Culurciello, Eugenio. (2017) *LinkNet: Exploiting Encoder Representations for Efficient Semantic Segmentation.* https://arxiv.org/pdf/1707.03718.pdf.

[9] Badrinarayanan, Vijay & Kendall, Alex & Cipolla, Roberto. (2016) *SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation.* https://arxiv.org/pdf/1511.00561.pdf.

[10] Long, Jonathan & Shelhamer, Evan & Darrell, Trevor. (2015) *Fully Convolutional Networks for Semantic Segmentation.* https://arxiv.org/pdf/1411.4038.pdf.

[11] He, Kaiming & Zhang, Xiangyu & Ren, Shaoqing & Sun, Jian. (2015) *Deep Residual Learning for Image Recognition.* https://arxiv.org/pdf/1512.03385.pdf

[12] Github repository with Keras implementation of LinkNet: https://github.com/davidtvs/Keras-LinkNet

[13] Weikang, Fan & Huiqin, Jiang & Ling, Ma & Jianbo, Gao & Haojin Yang *A modified faster R-CNN method to improve the performance of the pulmonary nodule detection.*

[14] S. Kido, Y. Hirano and N. Hashimoto. Detection and classification of lung abnormalities by use of convolutional neural network (CNN) and regions with CNN features (R-CNN). 2018 International Workshop on Advanced Image Technology (IWAIT), Chiang Mai, 2018, pp. 1-4.

[15] Github repository with implementation of Faster R-CNN https://github.com/facebookresearch/Detectron

[16] Atherosclerotic Vascular Calcification Detection and Segmentation on Low Dose Computed Tomography Scans Using Convolutional Neural Networks https://arxiv.org/pdf/1802.08717.pdf

[17] He, K. & Zhang, X. & Ren, S. & Sun, J. (2015) *Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification.* arXiv:1502.01852.